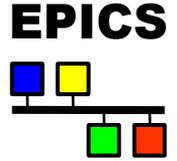


Save/Restore at the APS

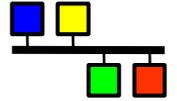
John Maclean
APS/ANL



Introduction



- Save and Restore, AKA bumpless reboot, at the APS
- History
- Highlights
- Usage
- Setpoint Restore
- Helpful features in 3.14
- Disclaimer: Tim Mooney, Frank Lenszkus and others did the work, I'm just a user



Original

Tim Mooney: 12/96
V1.3 → V1.5 Enumeration

Tim Mooney: 4/99
V1.6 → V2.3 Crash Protection

Frank Lenkszus: 10/97
Paths, passes.

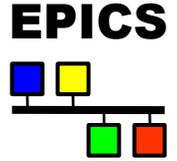
Tim Mooney: 2/02
V2.4 → V2.6 Macros

Available via EPICS web site:
In use on APS beamlines

Available internally:
In use on APS accelerator



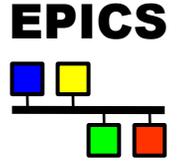
Highlights



- Save on; Monitor, trigger, period, manual
 - Monitor is the most useful
- Two restore passes, pre and post iocInit
- Settable request and save file directories
- Multiple request and save files (and directories)
- Backup files created
 - When save files saved “myFile.sav.bu”
 - When ioc reboots “myFile.sav.YYMMDD-HHMMSS”



Usage – preparation



- Build the source:

```
save_restore.c, dbrestore.c, initHooks.c, fGetDateStr.c,  
save_restore.h, fGetDateStr.h
```

- Create "request" files – e.g. myFile.req

- One PV per line
- Can include other request files
- Macro substitutions are performed
- e.g. inside myFile.req

```
myAI
```

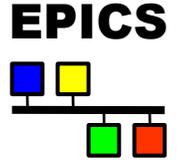
```
myAO
```

```
file myIncludedFile.req biscuits=cookies, drink=beer
```

- Save file will be created with .sav replacing .req i.e. myfile.sav



Usage – ioc startup commands



```
### Load custom EPICS software [including save_restore.o and dbrestore.o]
ld < xxxLib
.
.
# ok to restore a save set that had missing values (no CA connection to PV)?
sr_restore_incomplete_sets_ok = 1

# dbrestore saves a copy of the save file it restored.
reboot_restoreDatedBU = 1;

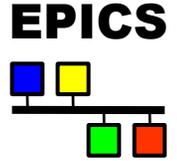
# specify where save files should go
set_savefile_path("myAppTop/mySaveDir");

# specify where request files come from
set_requestfile_path("myAppTop/myRequestDir")

# specify what save files should be restored when
set_pass0_restoreFile("myPositions.sav")
set_pass1_restoreFile("mySettings.sav")
```



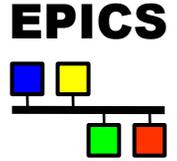
Usage – ioc startup cont.



```
# Currently, the only thing we do in initHooks is call reboot_restore().
# To disable autorestore, comment out the following line.
ld < initHooks.o
.
.
dbLoadDatabase("../../dbd/xxxApp.dbd")
dbLoadRecords("xxxApp/Db/whatever.db"...
.
.
iocInit
.
### Start up the autosave task and tell it what to do.
# save positions every five seconds
create_monitor_set("myPositions.req",5)
# save other things every thirty seconds
create_monitor_set("mySettings.req",30)
.
.
```



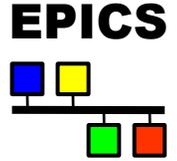
Where to find it



- There are many more features available in the utility than seen here
- See the README:
`http://www.aps.anl.gov/xfd/SoftDist/swBCDA/autoSaveRestore_3.13.5.README`
- Get the distribution:
`http://www.aps.anl.gov/xfd/SoftDist/swBCDA/autoSaveRestore_3.13.5.tar.gz`
- Even better, go via the EPICS web site to get make sure you get the up to date version:
`http://www.aps.anl.gov/epics/modules/soft.php`



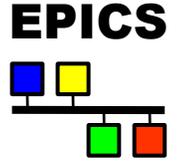
SetpointRestore



- For some power supplies bumpless reboot of ioc requires copying of actual DAC value in to setpoint PV without processing setpoint PV.
- Executed after iocInit.
- Takes list of source and destination PVs. e.g.
`S32A:Q1:DacAI S32A:Q1:CurrentAO`
`S32A:Q2:DacAI S32A:Q2:CurrentAO`
- Waits for given time before reading source PV.
`setpointRestore ("vpApp/psRestore/iocs31vpSPRestoreFile", 6)`
- Not available on EPICS web pages



Coming in 3.14



- Features to help save restore apps.
- Add info lines to record definitions in .db files e.g.

```
record (ai, "myAi") {  
    field(...)  
    ...  
    info(savePeriod, "10 seconds")  
    info(srGroup, "critical")  
    info(restore, "auto")  
    ...  
}
```

- A series of routines in dbStaticLib.c allow you to Get/Put the info settings etc.